Module 7 Exploratory Data Analysis Assignment

Your Tasks:

1. **Read data from ComboCovidTests.csv into Pandas dataframe named df_test. You should retrieve all columns.**
2. **Read data from zip_code_database.csv into Pandas dataframe named df_zip. You should only retrieve the Zip Code, Primary City, STATE, and IRS Estimated population columns.**
3. **Read data from covid_death_by_state.csv into Pandas dataframe named df_death. You should retrieve all columns.**

4. **Write a Python user-defined-function to clean the column headings of any Pandas dataframe:**

   def CleanColumnHeading(dfx):

   Your code to convert all column names to lower cases

   Your code to change all spaces in column names to underscore _

   return dfx

5. **Clean column headings of df_test, df_zip, df_death using your CleanColumnHeading function.**
6. **Make sure columns containing zipcode in all dataframes have the same column name of zip, columns containing city names in all dataframes have the same column name of city, columns containing state names in all dataframes have the same column name of state.**
7. **Write a function named CleanCovidTest(df_test) to perform the Module 4 Python Managing Data Practice Worksheet Tasks on data stored in df_test. This function should return df_test to the caller:**

Your Tasks:

1) Pass df_test to the function CleanCovidTest(df_test)
2) df_test contains the Covid test data of 100 Covid testing clinics collected on May 6, 2020. You want to determine if each clinic's test string is valid using isValidString(s) from assignment 1. If the clinic's test string is valid, you want to find out the total number of tests performed by the clinic, the total number of positive test cases, and the total number of negative test cases.
3) Add a column to df_test and name it "is_valid" to store the values "yes" for True and "no" for False based on the isValidString(s) results.
4) Add another column to df_test and name it "total_tests" to store the total number of tests performed by each clinic
5) Add another column to df_test and name it "positive" to store the total number of positive test cases for each clinic
6) Add another column to the data frame and name it "negative" to store the total number of negative tests cases for each clinic
7) Return the updated df_test to the caller

NOTE:

Use the SPECS for isValidString from Python Assignment 1. Use all functions you developed for Python Assignment 1 to solve this problem. You might need to fix your Python Assignment 1 code if you did not pass the 30 test cases we used to test your code.

8. Use Pandas merge function to merge df_test and df_zip into a new dataframe name df_new
9. Use Pandas merge function to merge df_new and df_death into df_new.
10. Build a filter for df_new to obtain:
    a. States that had the number of death greater than 10000. Store the result in a variable name death_result
    b. Cities that had the number of positive test results greater than 10. Store the result in a variable name pos_result
    c. Clinics that had the number of positive test results greater than 10 and death greater than 10000 and IRS Estimated Population greater than 30000. Store the result in a variable name pop_result

Save your program as mod7assignment.py. Your program should not have any outputs to the screen. You can use output statements to test your program, but you need to remove them before submitting the assignment. Your program should include these functions:

CleanColumHeading(dfx)

CleanCovidTest(df_test)

isValidString(s)

You can use all Python built-in and imported functions and features we covered in class.